Highly Accurate Switched Capacitor DAC

invented by:

Scott C. McLeod

Aniruddha Bashar

BACKGROUND OF THE INVENTION

<u>Field of the Invention</u>

This invention relates generally to the field of analog circuit design and, more particularly, to Digital to Analog Converter (DAC) design.

5

<u>Description of the Related Art</u>

Analog-to-digital converters (ADCs) are circuits used to convert signals from the analog domain, where the signals are typically represented by continuous quantities such as voltage and current, to the digital domain, where the signals are generally represented by discrete quantities such as numbers. Similarly, Digital-to-Analog converters (DACs) are circuits used to convert signals from the digital domain to the analog domain. These circuits can be implemented in a variety of ways. Well known and often used conversion techniques include flash, delta-sigma (or sigma-delta), sub-ranging, successive approximation, and integrating.

One often-utilized basic building block of an ADC is the analog integrator, commonly implemented as a switched-capacitor integrator (SCI) 100 illustrated in Fig. 1. Operation of SCI 100 consists of two main phases, the sampling phase and the charge transfer, or integration phase. During the sampling phase an input voltage v_i 110 is coupled to a first terminal of input capacitor C_{inp} 104 through switch P1 120, while switch P2 122 couples the second terminal of C_{inp} 104 to ground. Thus C_{inp} 104 is charged to a voltage level corresponding to v_i 110. During the integration phase, P1 120 is used to couple the first terminal of C_{inp} 104 to ground, while P2 122 is switched to couple the second terminal of C_{inp} 104 to the inverting input terminal of operational amplifier (OP-AMP) 102, and to integration capacitor C_{int} 106. C_{int} 106 is connected to form a feedback loop between the inverting input and output terminal of OP-AMP 102. Thus, during the integration phase the charge stored across C_{inp} 104 is transferred to C_{int} 106. The ratio C_{inp}/C_{int} determines the gain of v_o 112 with respect to v_i 110.

One example of an ADC (and DAC) is the "delta-sigma converter" or "sigma-delta converter", which is well known in the art. Use of Delta-sigma (D/S) converters

has proliferated due primarily to their capability for high-resolution analog-to-digital conversion in mixed signal VLSI processors. A D/S converter typically digitizes an analog signal at a very high sampling rate (multiple oversampling) in order to perform noise shaping. Digital filtering following the noise shaping allows the D/S converter to achieve a higher resolution than conventional ADCs. Decimation after the filtering reduces the effective sampling rate to the "Nyquist" rate.

Figure 2 illustrates a block diagram of a single bit D/S converter 10, commonly known in the art. Single bit D/S converter 10 includes a single bit D/S modulator 12 coupled to a digital filter and decimation circuit 14. D/S modulator 12 includes a summing node 16, an integrator 18, a single bit ADC 20, and a single bit DAC 22. DAC 22 is coupled to the output of ADC 20 and provides feedback to summing node 16. An analog input signal Vin is connected to one input (add) of summing node 16, and the output of DAC 22 is connected to another input (subtract) of summing node 16. In operation, the output of summing node 16 is integrated by integrator 18 and subsequently converted into a single bit, digital signal by ADC 20. The single bit digital signal is in turn converted back to an analog signal by DAC 22 and subtracted from analog input signal Vin at summing node 16. Single bit D/S modulator 12 converts Vin into a continuous serial stream of 1s and 0s at a rate determined by a sampling clock frequency, kf_S. Due to the feedback provided by DAC 22, the average value output by DAC 22 approaches that of Vin when the gain of the loop is high enough.

Figure 3 shows a block diagram of a multi-bit D/S converter 50. Multi-bit D/S converter 50 includes a multi-bit D/S modulator 52 coupled to a multi-bit digital filter and decimation circuit 54. Multi-bit D/S modulator 52 further includes a summing node 56, an integrator 58, a multi-bit ADC 60, and a multi-bit internal DAC 62. Multi-bit D/S modulator 50 of Figure 3 operates similarly to the single-bit D/S converter of Figure 2. The output of summing node 56 is integrated by integrator 58 and converted into a multi-bit digital signal by multi-bit internal ADC 60 operating at oversampling rate kf_S. Multi-bit DAC 62 is connected via a feedback loop between the output of the multi-bit ADC 60 and an input node of the summing node 56, whereby the analog signal output of DAC 62 is subtracted from analog signal input Vin. Again, the output of DAC 62 approaches that of analog input signal Vin as a result of the feedback. Digital filter and decimation

circuit 54 removes quantization noise shaped into the higher frequencies and re-samples the over-sampled digital signal at rate f_S.

Multi-bit D/S converter 50 of Figure 3 provides benefits over single bit D/S converter 10 of Figure 2. Namely, the multi-bit D/S converter 50 provides more resolution and less quantization noise for a given oversampling rate. Additionally, multi-bit D/S converter 50 is more stable than single bit D/S converters. However, multi-bit D/S converter 50 suffers from linearity errors introduced by internal multi-bit DAC 62. Single bit D/S converters on the other hand do not produce linearity errors. Linearity error is due to the inability of the multi-bit DAC to accurately translate a digital input value into an analog current or voltage. In other words, given a particular digital input, the resulting analog output of multi-bit internal DAC 62 approximates the digital value but does not exactly equal the digital value. In reality, the actual analog output differs from the digital input value by an amount equal to the linearity error.

Another type of ADC is the successive approximation register (SAR) converter illustrated in Fig. 4. The conversion technique used by a SAR converter is also referred to as bit-weighing conversion, where typically a comparator 72 is used to compare the applied analog input voltage Vin 80 against the output of an N-bit DAC 76. Using the DAC 76 output as a reference, the final converted (digital) result Dout 86 is approached as a sum of N weighting steps, in which each step corresponds to a single-bit conversion. At the beginning of the conversion process the SAR 74 bits are all initialized to zero. The most significant bit (MSB) of SAR 74 is then set to '1' (or high) and the voltage as represented by SAR 74 (and produced by DAC 76) is compared with Vin 80. A Vin 80 value lower than the voltage represented by SAR 74 would imply that SAR 74 holds too large a value, which has to be reduced, in which case the MSB of SAR 74 is reset to zero. On the other hand, a Vin 80 value higher than the voltage represented by SAR 74 would imply that the register value is not large enough to equal Vin 80, in which case the MSB of SAR 74 is allowed to retain its value of '1'. In the next cycle, the next significant bit of SAR 74 is set to '1' and the same process is performed iteratively. As each bit is determined, it is latched into SAR 74 as part of the ADC's output. Typically controlled by a logic control circuit 78 which is operated synchronously through the use of clock signal 82, the aforementioned steps are executed N times for an N-bit ADC, at the end of

which the contents of SAR 74 will correspond to the analog input voltage Vin 80 provided to the ADC. The beginning and end of the conversion process may be determined through a set of appropriate control signals.

Generally, single-bit DACs do not exhibit the non-linearity characteristics of multi-bit DACs. Accordingly, ADCs employing a single bit internal DAC do not suffer from linearity errors, and are therefore more accurate. In this respect, single bit internal DACs are preferred over multi-bit internal DACs. However, when utilizing the D/S technique, due to the resolution and stability of a multi-bit D/S converter being superior to that of a single bit D/S converter, it is preferable to use multi-bit D/S converters, where increased accuracy is achieved by removing or reducing the non-linearity produced by the D/S converter's internal multi-bit DAC. Similarly, while SAR converters generally operate at fast speeds and typically feature lower complexity and low power consumption, they are also directly affected by the accuracy of their internal DAC.

One technique used for increasing DAC accuracy has been Dynamic Element Matching (DEM), which cycles through a multiplicity of unit capacitors used in the DAC to cancel out mismatch errors. This technique typically requires a large silicon area because a unit capacitor is needed for each least significant bit (LSB) of the DAC. For example, a five-bit DAC would require 31 separate unit capacitors. Similarly, a 16-bit audio DAC would typically require ~65k unit capacitors. To make an accurate audio DAC, a Delta Sigma architecture is generally used with a multi-bit quantizer using the aforementioned DEM technique.

In other words, one drawback of the DEM technique is that its use is typically restricted to low bit DACs since it requires a unit capacitor for each LSB of the DAC. Present day audio DACs are generally designed using the D/S technique. Though systems with resolutions of up to 24-bits have apparently been achieved using this technique, the linearity of such systems is difficult to verify. The drawback of D/S based design lies in the complexity of the DEM technique itself and the complex analog output filter it typically requires. Previously, capacitor DAC accuracy has been limited to the physical matching obtainable on silicon, which is approximately 0.1% (11bits).

One proposed solution to the DAC accuracy problem has been the use of a technique of charge redistribution first introduced by Suarez et al in the IEEE publication

"All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques—Part II", IEEE Journal of Solid State Circuits, Vol. SC-10, No. 6, December 1975. The technique involves the use of two small capacitors and some switches to juggle a charge between the two capacitors to form the final DAC voltage. A significant limitation of this technique lies in its susceptibility to a mismatch error between the two capacitors. It also requires more clock cycles to convert the digital word into an analog voltage than other available techniques.

Therefore, there still exists a need for a system and method for designing highly accurate, low power, compact size DACs.

SUMMARY OF THE INVENTION

In one set of embodiments the invention comprises a system and method for designing and operating a highly accurate, low-power, compact size DAC. The charge redistribution technique first introduced by Suarez et al in the IEEE publication "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques—Part II", IEEE Journal of Solid State Circuits, Vol. SC-10, No. 6, December 1975, may be employed, where charge redistribution may be performed a first time and again a second time in a complementary fashion, followed by a summing of the two charge distributions, in effect canceling the odd order capacitor mismatch errors. In other words, two complementary conversions may be performed and added together to form a final DAC output voltage. The two complementary conversions, when added, may cancel all odd order mismatch errors such that the accuracy of the DAC may become a function of the square of the mismatch of the two capacitors, resulting in greatly increased accuracy. When performing the complementary conversions for multiple bits, the sequence in which each of the two capacitors is charged may be determined to minimize the even-order errors, especially second-order errors. The invention also proposes applying the DEM technique in conjunction with the complementary conversions, thus substantially reducing even-order errors in addition to eliminating odd-order errors, with less oversampling than required by current DEM implementations.

Without the proposed canceling, capacitors matched to 0.1% would allow for a resolution of 11 bits, while applying the proposed technique may result in capacitors matched to 0.1% allowing for resolutions of up to twenty-two bits. Integrating the DEM technique with the complementary conversion technique makes it possible to eliminate first, second, and third order errors of a DAC operating with eight times oversampling for up to six bits. D/S multi-bit quantizers may typically use DACs with resolution of less than or equal to six bits. This implies that an accuracy of forty-four bits may be possible with capacitors matching to 0.1%, and any error contributed by capacitor mismatch may be eliminated, though final DAC accuracy will be limited by other error sources in the DAC. The performance of D/S DACs and ADCs may thus also be greatly improved.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing, as well as other objects, features, and advantages of this invention may be more completely understood by reference to the following detailed description when read together with the accompanying drawings in which:

Fig. 1 illustrates a switched-capacitor integrator in accordance with prior art;

Fig. 2 illustrates a schematic diagram of one embodiment of a single bit D/S converter in accordance with prior art;

Fig. 3 illustrates a block diagram of one embodiment of a multi-bit D/S converter in accordance with prior art;

Fig. 4 illustrates a block diagram of a successive approximation register converter in accordance with prior art;

Fig. 5 illustrates a schematic diagram of one embodiment of a DAC in accordance with the present invention;

Fig. 6 illustrates a table of error-terms calculated for each bit of a 13-bit binary number to be converted in accordance with one set of embodiments of the present invention; and

Fig. 7 illustrates a tree diagram of possible paths to select when performing four first-pass/complementary switching sequences while minimizing and/or eliminating second-order and higher even-order errors, in accordance with one set of embodiments of the present invention.

While the invention is susceptible to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and will herein be described in detail. It should be understood, however, that the drawings and detailed description thereto are not intended to limit the invention to the particular form disclosed, but on the contrary, the intention is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of the present invention as defined by the appended claims. Note, the headings are for organizational purposes only and are not meant to be used to limit or interpret the description or claims. Furthermore, note that the word "may" is used throughout this application in a permissive sense (i.e., having the potential to, being able to), not a mandatory sense (i.e., must)." The term "include", and derivations thereof, mean "including, but not limited to". The term "coupled" means "directly or indirectly connected".

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

As used herein, when referencing a pulse of a signal, a "leading edge" of the pulse is a first edge of the pulse, resulting from the value of the signal changing from a default value, and a "trailing edge" is a second edge of the pulse, resulting from the value of the signal returning to the default value. A first signal is said to be "corresponding" to a second signal if the first signal was generated in response to the second signal. When data is said to be "registered" or "latched" "using" a signal, the signal acts as a trigger signal that controls the storing of the data into the register or latch. In other words, when a signal "used" for registering or latching data is in its triggering state, the data residing at respective input ports of the register or latch is stored into the register or latch. Similarly, when data is latched "on the leading edge" or "on the trailing edge" of a pulse of a clock, the data residing at respective input ports of a register or latch is stored into the register or latch, respectively, when a leading edge or a trailing edge of a pulse of the clock occurs, respectively. A first signal is said to "propagated based on" a second signal, when the second signal controls the propagation of the first signal. Similarly, a first module is said to "use" a clock signal to transfer data to a second module, when propagation of the data from the first module to the second module is controlled and/or triggered by the clock signal. When referencing a binary number, the least significant bit (LSB) is understood to be the rightmost bit of the binary number, whereas the most significant bit (MSB) is understood to be the leftmost bit of the binary number. For example, in case of the binary number '011' the LSB would be '1' while the MSB would be '0'.

Fig. 5 shows the schematic of a DAC built around a switched-capacitor integrator (SCI), operated in accordance with one set of embodiments of the present invention. In this embodiment, DAC 200 includes a charge redistribution circuit (CRC) 250 functionally configured and coupled as an input capacitor-unit of an SCI 260, which is coupled to a hold/output circuit 270. CRC 250 may include capacitors C_1 210 and C_2 212 coupled in parallel. Through node 220, one terminal of C_1 210 may be coupled to a reference voltage Vref 230 via switch S1, coupled to a common ground via switch S4,

and coupled to node 222 via switch S3. The other terminal of C_1 210 may be coupled to the common ground via switch S5. Similarly, through node 222, one terminal of C_2 212 may be coupled to Vref 230 via switch S2 and to the common ground via switch S7. The other terminal of C_2 212 may be coupled to the common ground via switch S6.

5   In one set of embodiments, capacitor C_1 210 may be selected to have substantially the same capacitance value as capacitor C_2 212. However, for example, due to process variation during fabrication a mismatch may exist between the values of C_1 210 and C_2 212. If C is the average value of C_1 210 and C_2 212, and Δ is defined as one-half of the difference between C_1 210 and C_2 212, then C_1 210 and C_2 212 may be
10   expressed as:

(1) $C_1 = C + \Delta C = C * (1+\Delta)$;

(2) $C_2 = C - \Delta C = C * (1-\Delta)$;

(3) $C_1 + C_2 = C + \Delta C + C - \Delta C = 2 * C$.

For example, if the value of C_1 210 matches the value of C_2 212 by 1%, then:

15   (4) $\Delta = \frac{1}{2} * 0.01 = 0.005$.

The DAC configured in Fig. 5 may be operated such that odd-order errors introduced by C_1 210 matching C_2 212 within 1% (i.e. C_1 210 not matching C_2 212 perfectly) are eliminated. The DAC configured in Fig. 5 may also be operated such that even-order errors are minimized.

20   Referring to Fig. 5, holding switches S8, S9, and S10 of SCI 260 open, calculating (converting) a single bit may be performed by first charging C_1 210 to V_{ref} 230 (resulting in C_1 210 holding a charge Q_1), and shorting C_2 212 to ground (resulting in C_2 212 not holding any charge) by closing switches S1, S5, S6, and S7, and keeping switches S2, S3, and S4 open. Subsequently, closing switch S3 and opening switches S1,
25   and S7 would result in Q_1 now being shared between C_1 210 and C_2 212, leading to C_1 210 holding a nominal charge $Q_1/2$, and C_2 212 holding a nominal charge $Q_1/2$. The charge held by C_1 210 may be expressed as:

(5)
$$\frac{V_{ref}C_1{}^2}{C_1 + C_2},$$

and the charge held by C_2 212 may be expressed as:

(6)
$$\frac{V_{ref}C_1C_2}{C_1 + C_2}.$$

Selecting integrator feedback capacitor (C_F) 214 to have a capacitance value substantially

5   equal to the combined capacitance value of C_1 210 and C_2 212, (in other words, to the

overall capacitance value observed between nodes 220 and 224 with switches S3, S8 and

S9 closed with all other switches held open, resulting from capacitor C_1 210 and C_2 212

being coupled in parallel) and transferring the charge $Q_1/2$ from C_1 210 to C_F 214 by

opening switches S3, S5, S6, S11 and S13, and closing switches S4, S9 and S12, the

10   following output voltage may be obtained at node 228:

(7a)
$$V_{out} = \frac{V_{ref}C_1{}^2}{(C_1 + C_2)*C_F}.$$

At this point, C_F 214 and $C_1 + C_2$ (the combined capacitance of C_1 210 coupled in parallel

with C_2 212) may be "flipped", that is the charge held by C_F 214 may be re-dumped onto

C_1 210 and C_2 212. This re-dumping of the charge may eliminate a need to match C_F 214

15   to the combination capacitance of $C_1 + C_2$, and the output voltage obtained at node 228

may be expressed as:

(7b)
$$V_{out} = \frac{V_{ref}C_1{}^2}{(C_1 + C_2)^2}.$$

It should be noted that for purposes of clarity and simplicity, when referencing the output

voltage at node 228 henceforth, a re-dumping of the charges from C_F 214 to C_1 210 and

20   C_2 212 as described above will be assumed, unless otherwise noted, to preferably express

the output voltage at node 228 using equation (7b).

Substituting into equation (7b) the values for C_1 and C_2 from equations (1) and (2), we

may obtain:

(8)
$$V_{out} = V_{ref} \frac{C^2(1+\Delta)^2}{(C+\Delta C+C-\Delta C)} = V_{ref} \frac{C^2(1+2\Delta+\Delta^2)}{(2C)^2} = V_{ref} \frac{1+2\Delta+\Delta^2}{4}.$$

Again referring to Fig. 5, while holding switches S8, S9, and S10 of SCI 260 open, C_2 212 may be charged to V_{ref} 230 (resulting in C_2 212 holding a charge Q_2), and C_1 210 may be shorted to ground (resulting in C_1 210 not holding any charge) by closing switches S2, S4, S5, and S6, and keeping switches S1, S3, and S7 open. Subsequently, closing switch S3 and opening switches S2, and S4 would result in Q_2 now being shared between C_1 210 and C_2 212, leading to C_1 210 holding a nominal charge $Q_2/2$, and C_2 212 holding a nominal charge $Q_2/2$. The charge held by C_1 210 may be expressed as:

(9a)
$$\frac{V_{ref} C_1 C_2}{C_1 + C_2},$$

and the charge held by C_2 212 may be expressed as:

(9b)
$$\frac{V_{ref} C_2{}^2}{C_1 + C_2}.$$

Charge $Q_2/2$ from C_2 212 is transferred to C_F 214 by opening switches S3, S5, S6, S11 and S13, and closing switches S7, S8 and S12, the following output voltage may be obtained at node 228:

(9c)
$$V_{out} = V_{ref} \frac{C^2(1-\Delta)^2}{(C+\Delta C+C-\Delta C)} = V_{ref} \frac{C^2(1-2\Delta+\Delta^2)}{(2C)^2} = V_{ref} \frac{1-2\Delta+\Delta^2}{4}.$$

The results of equations (8) and (9c) may be added and the resulting value of Vout is:

(9d)
$$V_{out} = V_{ref} \frac{1+2\Delta+\Delta^2}{4} + V_{ref} \frac{1-2\Delta+\Delta^2}{4} = V_{ref} \frac{1+\Delta^2}{2}.$$

Thus, final accuracy of the above bit may be represented by the term $1+\Delta^2$, which, for example, in case of a 0.1% matching of C_1 210 to C_2 212, yields $(1-0.0005^2) = 0.99999975$. In other words, using the charge distribution technique as described above, for capacitors C_1 210 and C_2 212 matching to 0.1%, the error introduced by the mismatch may be no more than 0.000025% contrasted with an error of 0.1% without using the

charge distribution technique. Furthermore, the process described above may be extended to multiple bits.

Conversion of a multiple-bit number may be accomplished by operating the DAC in Fig. 5 as follows. For purposes of illustration the binary number '101' will be used,

5   but it should be understood that any number comprised of any number of bits might be converted by following the method described herein.

As a default, each switch (S1 through S14) may be left in an open position unless otherwise specified. First, when converting the LSB '1', C_1 210 may be charged to V_{ref} by closing switches S1 and S5, and C_2 212 may be shorted to ground by closing switches

10   S6 and S7. Subsequently, switches S1 and S7 may be re-opened and switches S5 and S6 may remain closed. By then closing switch S3, the previously stored charge on C_1 210 would now be shared between C_1 210 and C_2 212. In other words, the value of the respective nominal voltage across both C_1 210 and C_2 212 would now be $V_{ref}/2$ if the value of C_1 210 were exactly matched to the value of C_2 212. The switching method

15   described herein seeks to correct the error that may be introduced by a mismatch present between C_1 210 and C_2 212 due, for example, to process variation during fabrication.

Switch S3 may now be opened in order to convert the next significant bit of the binary number. A determination may be made whether to charge C_1 210 or C_2 212. While, at this point, a choice of which capacitor (C_1 210 or C_2 212) to charge may not

20   have a direct effect on the elimination of odd-order errors, it may be significant when considering the minimization, or possible elimination, of even-order errors. A method for such selection to minimize even-order errors will be further described below. In this example C_2 212 may be selected as the next capacitor to receive a new charge. Therefore, in the instance of a bit value of '0' the current charge on C_1 210 may be

25   retained by keeping switch S1 and S4 open and switch S5 remaining closed, and C_2 212 may be discharged by closing switch S7 (note that switch S6 remains closed). Now opening switch S7 and closing switch S3 the charge currently residing on C_1 210 (corresponding to a voltage of $V_{ref}/2$ across C_1 210) would be shared between C_1 210 and C_2 212. As a result, the value of the respective nominal voltage across both C_1 210 and

30   C_2 212 would be $V_{ref}/4$.

For the following bit value of '1', C_2 212 may be selected and charged to V_{ref} by opening switches S3 and S7, and closing switches S2 (note that switch S6 remains closed). The current voltage of $V_{ref}/4$ may be preserved across C_1 210 by also opening switches S1 and S4. Once again, by opening switch S2 and closing switch S3, the total charge would be distributed between C_1 210 and C_2 212, resulting in a nominal voltage value of $5/8*V_{ref}$ across both C_1 210 and C_2 212. The total charge may be transferred to integration capacitor C_F 214 by first opening switches S5, S6, S11, and S13, closing switch S12, and closing either S8 and S7 or S9 and S4 depending on which capacitor, C_1 210 or C_2 212, is selected for the source from which to transfer the charge to C_F 214.

Once the charge has been transferred to C_F 214, switches S8 and S9 may be opened and the entire switching sequence for converting the binary number '101' may be repeated with a complementary selection of the capacitor to be charged for each respective bit of the binary number. For example, if during the original sequence C_1 210 was selected to be charged when converting the LSB of '101', this time C_2 212 may be selected, and so forth. If 'm + n' represents the number of bits in the binary number to be converted, an equation for the output voltage generated for each Nth bit at node 228 may be expressed as:

$$(10) \qquad V_{bit(N)} = V_{ref} * \frac{\left(C_1^n * C_2^m + C_1^m * C_2^n\right)}{\left(C_1 + C_2\right)^{(n+m)}},$$

where 'm' and 'n' also represent the number of times a selected capacitor is charged/discharged through the first-pass and then the complementary pass, respectively. For example, in the term ($C_1^n * C_2^m$), 'n' and 'm' also indicate that capacitor C_1 is charged 'n' times and capacitor C_2 is charged 'm' times, respectively, during the first-pass switching sequence. Similarly, in the adjoining term ($C_1^m * C_2^n$), 'm' and 'n' indicate that capacitor C_1 is charged, in a complementary fashion, 'm' times and capacitor C_2 is charged 'n' times during the complementary switching sequence.

Once completed and the resulting charge transferred to C_F 214, then C_F 214 and $C_1 + C_2$ (the combined capacitance of C_1 210 coupled in parallel with C_2 212) may be "flipped", that is the charge held by C_F 214 may be re-dumped onto C_1 210 and C_2 212,

and hence the corresponding output voltage at node 228 of OTA 202, will be free of all odd-order errors. As will be indicated further below, second-order errors (as well as other even-order errors) may be minimized, or in some cases eliminated, by selecting in a specific sequence the capacitors to be charged/discharged for each respective bit of the binary number to be converted.

The following equations represent a mathematical formulation of the charges appearing on C_1 210 and C_2 212, respectively, through applying a first-pass switching sequence followed by a corresponding complementary switching-sequence as described above. While, for purposes of illustration, the following analysis is performed for a four-bit binary number, those skilled in the art will appreciate that the analysis is in no way restricted to four-bit binary numbers and may be performed for a binary number of any length. Equations (11-1) through (11-8) and (12-1) through (12-8) represent the charges appearing on C_1 210 and C_2 212, respectively, corresponding to each respective step during the first-pass switching sequence. Similarly, equations (13-1) through (13-8) and (14-1) through (14-8) represent the charges appearing on C_1 210 and C_2 212, respectively, corresponding to each respective step during the complementary switching sequence. The index of a respective step is indicated by the second digit in the equation number – for example equation (11-3) specifies the charge on C_1 210 at the end of step '3' of the first-pass switching sequence and equation (14-4) specifies the charge on C_2 212 at the end of step '4' of the complementary switching sequence. During each odd-numbered step the respective capacitor is either charged to V_{ref} 230 or holds its charge from the previous step, and during each even-numbered step the total charge is shared between C_1 210 and C_2 212.

In the following equations, bit(0), bit(1), bit(2) and bit(3) represent the respective individual bits of the four-bit binary number for which the analysis is being performed, (bit(0) being the LSB and bit(3) being the MSB), where each bit may have a value of either '1' or '0'. C_1 and C_2 may be defined as listed in equations (1) and (2), respectively, leading to the sum 'C_1+C_2' being defined as listed in equation (3).

The charge on C_1 210 at the end of each respective step during the first-pass switching sequence may be represented as follows:

(11-1) $V_{ref} * C_1 * bit(0)$

(11-2) $V_{ref} * \left(\dfrac{C_1^2 * bit(0)}{(C_1 + C_2)} \right)$

(11-3) $V_{ref} * \left(\dfrac{C_1^2 * bit(0)}{(C_1 + C_2)} \right)$

(11-4) $V_{ref} * \left(\dfrac{C_1^3 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(1)}{(C_1 + C_2)} \right)$

5 (11-5) $V_{ref} * \left(C_1 * bit(2) \right)$

(11-6) $V_{ref} * \left(\dfrac{C_1^3 * C_2 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_1 * C_2^2 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_1^2 * bit(2)}{(C_1 + C_2)} \right)$

(11-7) $V_{ref} * \left(\dfrac{C_1^3 * C_2 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_1 * C_2^2 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_1^2 * bit(2)}{(C_1 + C_2)} \right)$

(11-8) $V_{ref} * \left(\dfrac{C_1^4 * C_2 * bit(0)}{(C_1 + C_2)^4} + \dfrac{C_1^2 * C_2^2 * bit(1)}{(C_1 + C_2)^3} + \dfrac{C_1^3 * bit(2)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(3)}{(C_1 + C_2)} \right).$

The charge on C_2 212 at the end of each respective step during the first-pass

10 switching sequence may be represented as follows:

(12-1) 0

(12-2) $V_{ref} * \left(\dfrac{C_1 * C_2 * bit(0)}{(C_1 + C_2)} \right)$

(12-3) $V_{ref} * \left(C_2 * bit(1) \right)$

(12-4) $V_{ref} * \left(\dfrac{C_1^2 * C_2 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_2^2 * bit(1)}{(C_1 + C_2)} \right)$

15 (12-5) $V_{ref} * \dfrac{C_1^2 * C_2 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_2^2 * bit(1)}{(C_1 + C_2)}$

(12-6) $\quad V_{ref} * \left(\dfrac{C_1^2 * C_2^2 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_2^3 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(2)}{(C_1 + C_2)} \right)$

(12-7) $\quad V_{ref} * \left(C_2 * bit(3) \right)$

(12-8) $\quad V_{ref} * \left(\dfrac{C_1^3 * C_2^2 * bit(0)}{(C_1 + C_2)^4} + \dfrac{C_1 * C_2^3 * bit(1)}{(C_1 + C_2)^3} + \dfrac{C_1^2 * C_2 * bit(2)}{(C_1 + C_2)^2} + \dfrac{C_2^2 * bit(3)}{(C_1 + C_2)} \right).$

The charge on C_1 210 at the end of each respective step during the complementary switching sequence may be represented as follows:

(13-1) $\quad 0$

(13-2) $\quad V_{ref} * \left(\dfrac{C_1 * C_2 * bit(0)}{(C_1 + C_2)} \right)$

(13-3) $\quad V_{ref} * \left(C_1 * bit(1) \right)$

(13-4) $\quad V_{ref} * \left(\dfrac{C_1 * C_2^2 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_1^2 * bit(1)}{(C_1 + C_2)} \right)$

(13-5) $\quad V_{ref} * \dfrac{C_1 * C_2^2 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_1^2 * bit(1)}{(C_1 + C_2)}$

(13-6) $\quad V_{ref} * \left(\dfrac{C_1^2 * C_2^2 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_1^3 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(2)}{(C_1 + C_2)} \right)$

(13-7) $\quad V_{ref} * \left(C_1 * bit(3) \right)$

(13-8) $\quad V_{ref} * \left(\dfrac{C_1^2 * C_2^3 * bit(0)}{(C_1 + C_2)^4} + \dfrac{C_1^3 * C_2 * bit(1)}{(C_1 + C_2)^3} + \dfrac{C_1 * C_2^2 * bit(2)}{(C_1 + C_2)^2} + \dfrac{C_1^2 * bit(3)}{(C_1 + C_2)} \right).$

The charge on C_2 212 at the end of each respective step during the complementary switching sequence may be represented as follows:

(14-1) $\quad V_{ref} * C_2 * bit(0)$

(14-2) $\quad V_{ref} * \left(\dfrac{C_2^2 * bit(0)}{(C_1 + C_2)} \right)$

(14-3) $\qquad V_{ref} * \left(\dfrac{C_2^2 * bit(0)}{(C_1 + C_2)} \right)$

(14-4) $\qquad V_{ref} * \left(\dfrac{C_2^3 * bit(0)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(1)}{(C_1 + C_2)} \right)$

(14-5) $\qquad V_{ref} * \left(C_2 * bit(2) \right)$

(14-6) $\qquad V_{ref} * \left(\dfrac{C_1 * C_2^3 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_1^2 * C_2 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_2^2 * bit(2)}{(C_1 + C_2)} \right)$

5 (14-7) $\qquad V_{ref} * \left(\dfrac{C_1 * C_2^3 * bit(0)}{(C_1 + C_2)^3} + \dfrac{C_1^2 * C_2 * bit(1)}{(C_1 + C_2)^2} + \dfrac{C_2^2 * bit(2)}{(C_1 + C_2)} \right)$

(14-8) $\qquad V_{ref} * \left(\dfrac{C_1 * C_2^4 * bit(0)}{(C_1 + C_2)^4} + \dfrac{C_1^2 * C_2^2 * bit(1)}{(C_1 + C_2)^3} + \dfrac{C_2^3 * bit(2)}{(C_1 + C_2)^2} + \dfrac{C_1 * C_2 * bit(3)}{(C_1 + C_2)} \right).$

The output voltage at node 228 may then be represented in terms of the charge present on C_2 212 at the end of the first-pass switching sequence, specified in equation (12-8) and labeled as $Q_2(fp)$, and the charge present on C_1 210 at the end of the 10 complementary switching sequence, specified in equation (13-8) and labeled as $Q_1(cp)$:

(15) $\qquad V_{out} = \left(Q_2(fp) + Q_1(cp) \right) * \dfrac{1}{(C_1 + C_2)}.$

Substituting the values of $Q_2(fp)$ and $Q_1(cp)$ from equations (12-8) and (13-8), respectively, into equation (15), V_{out} may be expressed as:

(16) $\quad V_{out} = V_{ref} * \left(\dfrac{(C_1^2 * C_2^3 + C_1^3 * C_2^2) * bit(0)}{(C_1 + C_2)^5} + \dfrac{(C_1^3 * C_2 + C_1 * C_2^3) * bit(1)}{(C_1 + C_2)^4} + \dfrac{(C_1 * C_2^2 + C_1^2 * C_2) * bit(2)}{(C_1 + C_2)^3} + \dfrac{(C_1^2 + C_2^2) * bit(3)}{(C_1 + C_2)^2} \right).$

15 Substituting the values of C_1 210 and C_2 212 as defined in equations (1) and (2), respectively, into equation (16), V_{out} may be re-written as:

(17) $\quad V_{out} = V_{ref} * \left(\dfrac{2 * C^5 * (1 - 2 * \Delta^2) * bit(0)}{(2 * C)^5} + \dfrac{2 * C^4 * (1 - \Delta^4) * bit(1)}{(2 * C)^4} + \dfrac{2 * C^3 * (1 - \Delta^2) * bit(2)}{(2 * C)^3} + \dfrac{2 * C^2 * (1 + \Delta^2) * bit(3)}{(2 * C)^2} \right),$

and simplified as:

(18) $$V_{out} = V_{ref} * \left(bit(0) * \left(\frac{1}{16} - \frac{\Delta^2}{8} \right) + bit(1) * \left(\frac{1}{8} - \frac{\Delta^4}{8} \right) + bit(2) * \left(\frac{1}{4} - \frac{\Delta^2}{4} \right) + bit(3) * \left(\frac{1}{2} + \frac{\Delta^2}{2} \right) \right).$$

The results expressed in equation (18) indicate an absence of odd-order errors with error terms of only second-order or above present. For a ±1% matching between capacitors C_1 210 and C_2 212, the accuracy of the bits may be expressed using Δ as defined in equation (4), resulting in:

(19) $$\frac{\Delta^2}{2} = \frac{(\pm 0.005)^2}{2} = 0.0000125 * Fullscale,$$

which represents a greater than sixteen-bit accuracy.

For a ±0.1% matching between capacitors C_1 210 and C_2 212, we may write:

(20) $$\Delta = \pm 0.0005,$$

and

(21) $$\frac{\Delta^2}{2} = \frac{(\pm 0.0005)^2}{2} = 0.000000125 * Fullscale,$$

which represents a greater than twenty-three-bit accuracy. For an eleven-bit DAC the output voltage at node 228 may then be expressed as:

(22) $$V_{out} = V_{ref} * \left(bit(0) * \left(\frac{1}{2048} - \frac{\Delta^2}{512} \right) + bit(1) * \left(\frac{1}{1024} - \frac{5*\Delta^4}{1024} \right) + bit(2) * \left(\frac{1}{512} - \frac{3*\Delta^2}{512} \right) + bit(3) * \left(\frac{1}{256} + \frac{\Delta^2}{64} \right) \right) +$$

$$V_{ref} * \left(bit(4) * \left(\frac{1}{128} - \frac{\Delta^2}{64} \right) + bit(5) * \left(\frac{1}{64} - \frac{3*\Delta^2}{64} \right) + bit(6) * \left(\frac{1}{32} - \frac{\Delta^2}{32} \right) + bit(7) * \left(\frac{1}{16} - \frac{\Delta^2}{8} \right) \right) +$$

$$V_{ref} * \left(bit(8) * \left(\frac{1}{8} - \frac{\Delta^4}{8} \right) + bit(9) * \left(\frac{1}{4} - \frac{\Delta^2}{4} \right) + bit(10) * \left(\frac{1}{2} + \frac{\Delta^2}{2} \right) \right).$$

In one embodiment, operation of switches S1 through S14 in the DAC of Fig. 5 may require four clock pulses per bit to be converted. These clock pulses may be very fast as no amplifier settling may need to be taken into account, only the sharing of charge between C_1 210 and C_2 212, and the charging of C_1 210 and C_2 212 to reference voltage V_{ref} 230, respectively. At the end of each pass (entire switching sequence for the entire binary number to be converted), the charge from either C_1 210 or C_2 212 may be dumped

to C_F 214. At the end of the complementary switching sequence C_F 214 and $C_1 + C_2$ (C_1 210 in parallel with C_2 212) may be "flipped" as previously described during discussion of equation (7b). That is, the summed first-pass/complementary-pass charges may be re-dumped from C_F 214 onto C_1 210 and C_2 212. Following the re-dump, the resulting voltage at node 228 may be transferred to hold capacitor C_H 216 where it may be held as long as desired. In one embodiment, seven more cycles are added to the total conversion time to account for the charge re-dump and transfer to C_H 216, and autozeroing of amplifiers 202 and 204. In this embodiment, the conversion time for an N-bit binary number may be expressed as:

(23) $$T_{conversion} = \frac{4*N+7}{F_s},$$

where F_s is the sampling clock frequency.

Referring now to equation (10), it is apparent that multiple combinations of 'm' and 'n' may be considered for the converting of each bit. Considering Δ as previously defined, the error terms associated with each combination of 'm' and 'n' for each bit may be calculated. Substituting the terms for C_1 and C_2 from equations (1) and (2), respectively, into equation (10), the voltage for an Nth bit may be expressed as:

(24) $$V_{bit(N)} = V_{ref} * \left(\frac{(1+\Delta)^n (1-\Delta)^m + (1+\Delta)^m (1-\Delta)^n}{2^{n+m}} \right).$$

Each bit may be formed with different values of 'n' and 'm', where the bit number N corresponds to $m + n - 1$. In this case, since the MSB is most affected by the mismatch between C_1 210 and C_2 212, the error terms may be calculated not in the order in which the bits are converted but in the order of the MSB to the LSB, with bit 1 designated as the MSB.

Fig. 6 shows a table that includes error terms calculated for bit 1 through bit 13, where bit 1 is the MSB, for all the pertinent combinations of 'm' and 'n'. Only errors up to the fourth-order term are shown, as the contribution of higher than fourth-order terms to the overall error may be substantially negligible. Although the error terms for any bit, or bits, subsequent to bit 13 are not presented in the table of Fig. 6, they may also be

calculated, and error terms for any number of desired bits may be obtained from equation (24). When converting an N-bit number, each respective bit - from bit 1 to bit N - may be formed by performing the first-pass switching sequence and the complementary switching sequence for one pair of 'n' and 'm' values (m,n) for the respective bit.

5 Selection of (m,n) for each subsequent bit may be selected such that either the value of 'm' or the value of 'n' changes by one from the previous bit. For example, when converting a 3-bit number, (m,n) may be (1,1) for bit 1, (2,1) for bit 2, and (2,2) for bit 3. The total error associated with the final voltage would be a sum of the three respective error terms associated with each (m,n) for each respective bit. In the case of the above

10 cited example of the 3-bit number, the error would be:

$$(25) \qquad Error = \frac{1}{2}*\left(-\Delta^2\right)+\frac{1}{4}*\left(-\Delta^2\right)+\frac{1}{8}*\left(-2*\Delta^2+\Delta^4\right).$$

Error terms will only be incurred for bit values of '1'. In other words, as is evident from the sets of equations (11-1) – (11-8), (12-1) – (12-8), (13-1) – (13-8) and (14-1) – (14-8), for any bit value of '0', the respective voltage term, and hence error term, would be zero.

15 Considering the error terms as shown in Fig. 6, a path may be determined from each bit to the next, where the total accrued error may be minimized. Following a particular path in switching the charges, it is possible to have the second-order terms in the individual error terms shown in Fig. 6 to cancel, when a combination of paths are added together. In other words, a combination of paths may be identified and added

20 together, such that the second-order error terms associated with each chosen path cancel each other when added together. Referring again to Fig. 6, when considering bit 1, if a first-pass/complementary pass switching sequence is performed four times, twice with (m,n) chosen as (1,1) and twice with (m,n) chosen as (2,0), the error (up to, but not including fourth-order and higher even-order error terms), corresponding to bit 1 adds up

25 to:

$$(26) \qquad Error_{bit1} = \frac{1}{2}*\left(\Delta^2\right)+\frac{1}{2}*\left(-\Delta^2\right)+\frac{1}{2}*\left(-\Delta^2\right)+\frac{1}{2}*\left(\Delta^2\right)=0.$$

As can be observed in equation (26), there are no second-order error terms present in the final result. Similarly, if (m,n) combinations for bit 2 are selected to be (2,1) three times

and (3,0) once, the error (up to, but not including fourth-order and higher even-order error terms), corresponding to bit 2 adds up to:

$$(27) \qquad Error_{bit2} = 3 * \frac{1}{4} * \left(-\Delta^2\right) + \frac{1}{4} * \left(3 * \Delta^2\right) = 0.$$

Again, there are no second-order errors terms present in the final sum of equation (27).

5   Based on the above analysis, a combination of paths may be obtained for converting an N-bit binary number, where each combination of paths leads to a canceling, or minimizing, of second-order errors and/or higher even-order errors. Considering bit 1 and bit 2, and the first-pass/complementary switching sequences being performed four times, a selected path combination for bit 1 and bit 2 may include a first

10   and second path that each include a (m,n) pair of (1,1) for bit 1 followed by a respective (m,n,) pair of (2,1) for bit 2, a third path that includes a (m,n) pair of (2,0) for bit 1 followed by a third (m,n) pair of (2,1) for bit 2, and finally, a fourth path that includes a second (m,n) pair of (2,0) for bit 1 followed by a (m,n) pair of (3,0) for bit 2. Since the selection of four paths for bit 1 and bit 2, respectively, coincide with the combinations set

15   forth above leading to the results of equations (26) and (27), the second-order error terms will sum to zero for both bit 1 and bit 2, as previously indicated in equations (26) and (27), respectively. The selection of paths may be extended to subsequent bits following the same considerations.

     Fig. 7 shows a tree diagram outlining a possible arrangement of four path

20   sequences when performing four combinations of switching in a manner that leads to the cancellation of second-order errors for as many bits as possible. Each box may contain a (m,n) path combination for the respective bit, and a multiplier indicating the number of times that path combination may be employed for the respective bit. While the diagram only extends the sequence to thirteen bits, it may be implemented for as many bits as

25   desired, starting with the MSB at the top. As previously mentioned, the paths may start at the MSB due to errors associated with the MSB having the greatest effect on a final voltage value, where the final voltage value is the final result of the digital to analog conversion of the selected N-bit binary number. The errors associated with each bit may be calculated while traversing the paths as illustrated in Fig. 7. For each bit, the (m,n)

pair(s) given in Fig. 7 may be cross-referenced to the respective error terms shown in Fig. 6, and added according to the multiplier as also indicated in Fig. 7. The errors for the first seven bits - after having performed all four combinations of first-pass/complementary switching sequences – are shown below, with the exception of bit 1 and bit 2, which are shown in equations (26) and (27).

$$(28) \quad Error_{bit3} = 4 * \frac{1}{8} * \left(-\Delta^4\right) = -\frac{1}{2} * \Delta^4 .$$

$$(29) \quad Error_{bit4} = 2 * \frac{1}{16} * \left(-2 * \Delta^2 + \Delta^4\right) + 2 * \frac{1}{16} * \left(2 * \Delta^2 - 3 * \Delta^4\right) = -\frac{1}{4} * \Delta^4 .$$

$$(30) \quad Error_{bit5} = \frac{1}{32} * \left(\left(-3 * \Delta^2 + 3 * \Delta^4\right) + 2 * \left(-\Delta^2 - \Delta^4\right) + \left(5 * \Delta^2 - 5 * \Delta^4\right)\right) = -\frac{1}{8} * \Delta^4 .$$

$$(31) \quad Error_{bit6} = \frac{1}{64} * \left(\left(-3 * \Delta^2 + 3 * \Delta^4\right) + 3 * \left(\Delta^2 - 5 * \Delta^4\right)\right) = -\frac{3}{16} * \Delta^4 .$$

$$(32) \quad Error_{bit7} = \frac{1}{128} * \left(3 * \left(-2 * \Delta^2\right) + \left(4 * \Delta^2 - 10 * \Delta^4\right)\right) = -\frac{1}{64} * \Delta^2 - \frac{5}{64} * \Delta^4 .$$

As may be observed from equations (26) through (32), by using four combinations of first-pass/complementary switching, first, second, and third order errors may be canceled for up to six bits. Bit 7 does incur a second-order error term, as shown in equation (32). However, for C_1 210 matching C_2 212 within 1%, a matching percentage that may be obtained without substantial difficulties during fabrication, the second-order error shown in equation (32) may calculate to a value of 0.00000039, which represents –128dB with respect to fullscale, and corresponds to approximately 21 bits of accuracy.

A six-bit (or less) DAC may be considered for a multi-bit quantizer in a delta-sigma ADC. A typical five-bit quantizer may require an accurate five-bit DAC. The linearity of a delta-sigma modulator may be no better than the linearity of its internal DAC, which may indicate that a best linearity obtainable with a standard switched-capacitor based DAC may be approximately ten bits, as the capacitors might not possibly be matched to obtain a better linearity. When utilizing DEM to alleviate this problem, thirty-two unit capacitors may be needed for a five-bit DAC, essentially leading to what may be considered thirty-two one-bit DACs. A method of switching for randomizing or

noise shaping the DACs may be complex, and the amount of required oversampling may be constrained to be greater than the number of one-bit DACs employed. In such a case an oversampling ratio of at least 32 might be needed for a five-bit quantizer.

Thus, various embodiments of the systems and methods described above may facilitate design and operation of a DAC with eight times oversampling for up to six bits without first, second and third-order errors. An accuracy of forty-four bits may be possible with capacitors matching to 0.1%, and any error contributed by capacitor mismatch may be eliminated, greatly improving the performance of delta-sigma DACs and ADCs.

Although the embodiments above have been described in considerable detail, other versions are possible. Numerous variations and modifications will become apparent to those skilled in the art once the above disclosure is fully appreciated. It is intended that the following claims be interpreted to embrace all such variations and modifications. Note the section headings used herein are for organizational purposes only and are not meant to limit the description provided herein or the claims attached hereto.